IMAGE PROCESSING ON LINE

Published in Image Processing On Line on 2020–02–22.
Submitted on 2019–01–26, accepted on 2020–01–07.
ISSN 2105–1232 © 2020 IPOL & the authors CC–BY–NC–SA
This article is available online with supplementary materials,
software, datasets and online demo at
https://doi.org/10.5201/ipol.2020.247

# Matching of Weakly-Localized Features under Different Geometric Models

Erez Farhan

EE, BGU, Israel
farhan@post.bgu.ac.il

## Abstract

Matching corresponding local patches between images is a fundamental building block in many computer-vision algorithms, reducing the high-dimensional challenge of recovering geometric relations between images to a series of relatively simple and independent tasks. This approach is geometrically very flexible and has clear computational advantages over more convoluted global solutions. But it also has two major practical shortcomings: 1) Sparsity: the need to rely on high-quality repeatable features for matching drives current local methods to discard low-textured image locations and leave them unanalysed; 2) Reliability: the limited spatial context in which those methods work often does not contain enough information for achieving reliable matches. In this work, we target a major blind spot of local feature matching: ill-textured locations. We observe that while classic methods avoided using poorly localized features (e.g. edges) as matching candidates, due to their low reliability, these features contain highly valuable information for image registration. We show how, given the appropriate geometric context, reliable matches can be produced from these features, contributing to a better coverage of the scene. We present a statistically attractive framework for encoding the uncertainty that stems from using weakly localized matches into a coupled geometric estimation and match extraction process. We examine the practical application of the proposed framework to the problems of guided matching and affine region expansion and show significant improvement over preceding methods.

## Source Code

The source code and documentation are available from the web page of this article[1]. The code is mainly a Matlab code that requires some Matlab toolboxes detailed in the ReadMe.txt file attached to the source code. Specific instructions on how to run the code, including some other dependencies, are also found in the ReadMe.txt file.

**Keywords:** local matching; affine transformation; perspective transformation; dense matching; registration

[1] https://doi.org/10.5201/ipol.2020.247

# 1 Introduction

Image registration is a fundamental problem in computer vision that has been consistently addressed in research during the last decades. This work focuses, in all its parts, on the common case of registration between two 2-D RGB images (see Figure 1), where we seek for the 2-D correspondence field (with abuse of notation)

$$\overrightarrow{F_{x,y}} = (F_x, F_y) \mid I_1(u,v) \quad \text{``}=\text{''} \quad I_2(u + F_x(u,v), v + F_y(u,v)) \quad \forall (u,v) \in \Omega_{\overrightarrow{F}},$$

where $I_1$,$I_2$ are two RGB images that share different projections of the same 3-D surfaces, and $\Omega_{\overrightarrow{F}}$ is the domain of $\overrightarrow{F_{x,y}}$. By "$=$", we mean that both sides of the equation are a projection of the same 3-D patch in every point, and not necessarily the same RGB level. We note that in many cases, $\overrightarrow{F_{x,y}}$ is not defined in all of $I_1$ due to common scene or viewpoint variations like occlusions, non-rigid motions or even zoom. We also note that $\overrightarrow{F_{x,y}}$ does not fully represent the geometric relation between the images, as $I_2$ is not necessarily contained in the *range* of $\overrightarrow{F_{x,y}}$. For this purpose, we can similarly define the reciprocal

$$\overrightarrow{F'_{x,y}} = (F'_x, F'_y) \mid I_2(u,v) \quad \text{``}=\text{''} \quad I_1(u + F'_x(u,v), v + F'_y(u,v)) \quad \forall (u,v) \in \Omega_{\overrightarrow{F'}}.$$

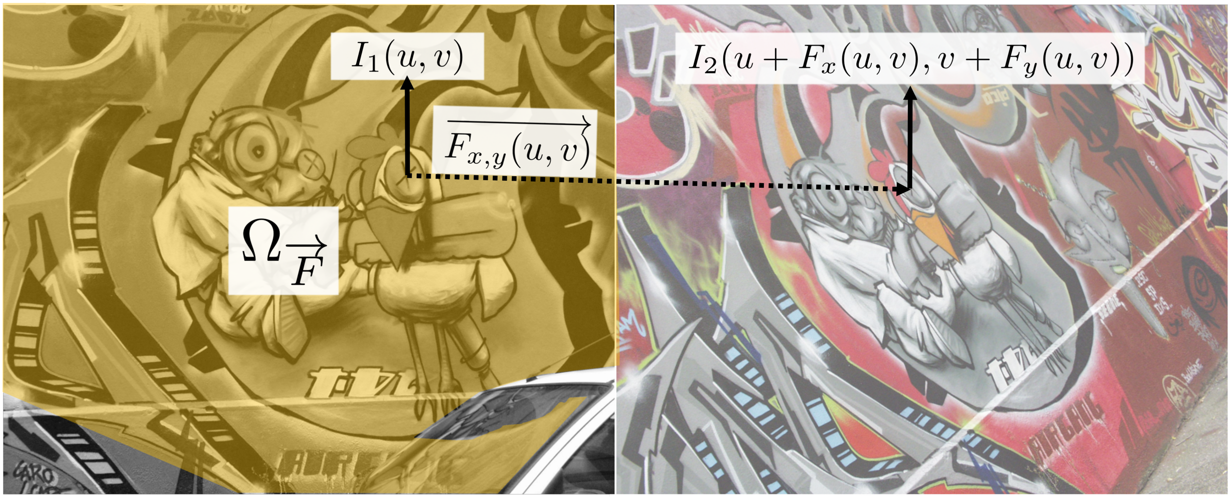

$I_1(u,v)$
$I_2(u + F_x(u,v), v + F_y(u,v))$
$\overrightarrow{F_{x,y}(u,v)}$
$\Omega_{\overrightarrow{F}}$

Figure 1: Illustrating the goal of image registration. The domain $\Omega_{\overrightarrow{F}}$ is well defined only in parts of the source image (yellow). $I_1(u,v) \quad \text{``}=\text{''} \quad I_2(u + F_x(u,v), v + F_y(u,v))$ doesn't necessarily represent equality in RGB levels.

Combining $\overrightarrow{F_{x,y}}$ and $\overrightarrow{F'_{x,y}}$ fully represents the geometric relations between the images with significant redundancy. Since $\overrightarrow{F_{x,y}}$ and $\overrightarrow{F'_{x,y}}$ can be analyzed with the same techniques, we focus our further discussion only on $\overrightarrow{F_{x,y}}$ for clarity of the exposition. As accurately estimating $\overrightarrow{F_{x,y}}$ in arbitrary scenarios has proven to be an extremely challenging task [14], many common scenarios such as small displacements, calibrated stereo or purely planar scenes allow significant simplifying assumptions on $\overrightarrow{F_{x,y}}$ [14], and encourage the search for compelling global solutions [26, 12, 22, 11] for estimating $\overrightarrow{F_{x,y}}$ in every point of the image. In contrast, local image matching is an attempt to *independently* estimate $\overrightarrow{F_{x,y}}$ in limited domains, where its behavior is relatively simple and the radiometric variations can be easily compensated [15, 2, 4, 19]. This divide-and-conquer nature carries both functional and computational advantages of the local approach over the global one in general scenarios, while also being better suited to cases where the domain $\Omega_{\overrightarrow{F}}$ is much smaller than the image, making global analysis irrelevant or wasteful. On the other hand, the strong dependency on local texture and the lack of context, still prevents local methods from achieving compelling coverage while maintaining high matching precision. In this work, we focus on increasing the coverage of local matching techniques while maintaining the high reliability achieved in preceding state-of-the-art work.

## 1.1 Local Matching

The classic process of finding local matches comprises four stages:

1. Extraction - Finding potential candidate points and defining small image patches around them (features).

2. Description - Attaching each feature with a descriptor [15, 2, 4] that codes the image within the patch.

3. Matching - Applying some metric to find potential matches between the descriptors.

4. Verification - Optionally applying higher-level considerations to reject potentially false matches.

While stages 1 and 2 are carried out independently between the images, information from different images is combined only in stage 3. The inherent challenge in this approach is that the image of a patch can vary dramatically as a result of geometric and radiometric variations. Failing to take these into account typically results in failure to “crop” repeatable features (stage 1) and robustly describe them (stage 2). This makes stage 3 very error-prone. Thus, a great body of work was directed towards extracting local features in an invariant or co-variant fashion and extracting robust descriptions of them [15, 18, 2]. Stage 4 might be very helpful in rejecting false matches, by applying some known physical prior on a specific setup [8]. While not always applicable, this stage is also less useful in situations of low signal-to-noise ratio, due to its commonly combinatorial nature. In practice, since local matching is inherently based on analyzing relatively small bits of information, the entire process is still prone to errors and inaccuracies. This led researchers to tighten the criteria and threshold for each of the stages. As a result of this tightening, there is constant tension between the theoretical advantages of local matching and its practical use in three major realms:

1. Sparsity - Reducing to highly repeatable and robustly describable features significantly reduces the set of matcheable locations, leading to sparse analysis of the scene and low coverage of $\Omega_{\vec{F}}$.

2. Computational Efficiency - Detection and description of these special features incurs a large computational demand.

3. Flexibility - Resorting to global information to reject outliers compromises some of the geometrical flexibility and model-free nature of local matching.

In this work, we tackle the problem of local feature sparsity by allowing accurate analysis of ill-textured features that are usually discarded by current matching methods due to their inherent uncertainty. We directly encode that uncertainty into the transformation estimation and match detection procedures. We then present a joint estimation and detection framework to fully utilize the information contained in these features and show how this procedure dramatically increases the coverage of $\Omega_{\vec{F}}$.

## 1.2 Geometric Transformation Estimation

Estimating geometric transformations between images is a key stage for many computer-vision algorithms. In general, the geometric connection between images is determined by the arbitrary structure of the scene in every point, its possible variation across time, and the poses from which the images were taken with respect to that scene. This makes image registration an arbitrarily high-dimensional problem and very challenging to tackle in a general fashion. Under specific setups, such as planar scenes, distant footage, or degenerate camera motions, low-rank geometric models can relate large common parts between images [14]. In other cases, such as rigid scenes or synchronized stereo acquisition, low-rank geometric models fully encode the epipolar geometry between the images, or between images and a 3D scene representation [14, 3, 25]. The most common approach for estimating such low-rank geometric entities involves initial extraction of geometric constraints, such as point or region matches [14, 3], followed by the incorporation of these constraints for model estimation. In general, there are several major factors affecting the accuracy of transformation estimations, common to all geometric models [14]: 1) the quality of local constraints; 2) the outlier handling technique; 3) the error minimization target. This works focuses mainly on (1) and its coupling to the estimation process, while also providing analytic derivations for a more statistically sound outlier rejection procedure.

## 1.3 Increasing the Pool of Matchable Local Features

Extracting large amounts of well-localized features in general scenarios has been a key challenge in computer-vision in the last decades [15, 16], trying to generally overcome image variations coming from different sources such as camera pose, scene dynamics or illumination conditions, and produce robust and repeatable features [18]. Thus, many robust feature extraction methods have been developed [16, 15, 13], and proved to successfully handle such variations in many different scenarios. This robustness is achieved at a twofold price: 1) higher computational demand - especially in higher resolution scenarios; 2) sparsity - where extracting only high-quality features leads to a very sparse feature population that is strongly dependent on the specific scene texture. For many scenarios and applications, these sparse local features lead to sufficient geometric estimation accuracy. In other cases, demanding higher estimation accuracy [27], denser coverage [20], or lacking appropriate texture (e.g. indoor walls), increasing the amount of reliably matched features can be highly beneficial. For this purpose, guided-matching methods have been developed [17, 9, 10], utilizing initially estimated geometrical transformations based on sparse repeatable features to locate a denser population of well-localized matches from less repeatable features unattainable by robust methods. A necessary condition for well-localized features is having a non-degenerate structure tensor [13]. Degenerate textures such as edges and smooth patches have been largely discarded as potential feature matches in general scenarios, due to their inherent localization uncertainty. In this work, we exploit the coupling between geometric estimation and local matching, to both reduce the localization uncertainty of degenerate features, and utilize the information stored in them to improve the estimation accuracy. In this context, this work can be viewed as an extension for guided-matching that goes beyond well-localized features to further increase the pool of matchable features (as illustrated in Figure 2).

## 1.4 Contribution to Affine-Expansion

The predication and correction mechanism utilized in this work highly resembles that of [6, 7], where local affine transformations around initial point matches are refined to extract many surrounding point matches using normalized cross-correlation (NCC) scans. For ill-textured patches, like those illustrated in Figure 3(a), this might carry a significant localization ambiguity. In [6], this ambiguity was tackled by filtering out all the poorly localized NCC results, and keeping only the results that

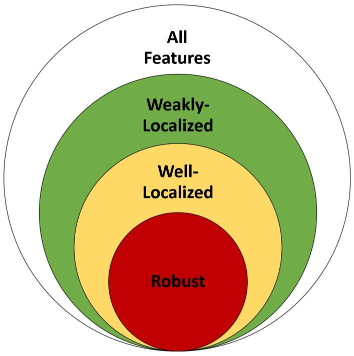


Figure 2: The local feature hierarchy: While guided-matching extends the matchable feature population from Robust only to all the Well-Localized ones, this work aims to also include the vast population of Weakly-Localized matches.

resemble a delta function (the "Delta-Criterion"), as illustrated in Figure 3(b). It was shown that this choice is preferable to utilizing all the scanned matches in a brute-force manner. We claim that the main shortcoming of the Delta-Criterion is in the choice to discard weakly-localized matches that contain significant information (e.g. edges). In this work, we directly show how to utilize these weakly-localized matches in the context of affine expansion and compare our results to the Delta-Criterion-based approach.

(a) Matching an edge-like patch yields ambiguous matching (strong responses are poorly localized)

(b) Matching a corner-like patch yields accurate matching (strong responses are well localized)

Figure 3: Example of the "delta" criterion in two cases: the strongest correlation responses are in dark red.

## 2 Matching under the Affine Model: Local Matching of Weakly-Localized Features

The 2D affine model is given by $Q(x) = [A]_{2\times2} \cdot [x]_{2\times1} + [b]_{2\times1}$, where $A$ is a non-singular matrix. The simplicity of this model (only 6 DOF), and its applicability in locally approximating arbitrary (smooth) geometric relations, have made it the model of choice for solving challenging image registration problems [19, 5, 18, 1]. In this section, we show how to exploit the applicability of the affine model for locating large amounts of point matches which are normally out of reach for standard matching techniques. Without significantly limiting the generality, we assume we are dealing with two images fully or partly related by a piece-wise smooth geometric transformation. To simplify the exposition, we focus on a single local region, centered at point $O$ in the source image, where we have an initially estimated affine transformation ($\hat{Q}(x)$) between the images. Much like [6], our goal is to locate more image matches in the vicinity of this local region. The steps for reaching this goal are detailed in the following sub-sections.

## 2.1 Step 1: Key-Point Extraction and Assignment

To locate new matches, we wish to extract candidate locations (key-points) in the vicinity of $O$ that can be successfully matched to the target image. We claim that under a known geometric model, and with the right treatment, weakly-localized features (i.e. with degenerate structure-tensor) can suffice for successful point matching. For simplicity, we opt for an elementary feature extraction method inspired by [23]. For each pixel within radius $R$ of $O$, we observe the *structure-tensor* defined by aggregating the derivatives in some $U \times V$ neighborhood around it

$$S \triangleq \begin{pmatrix} \Sigma I_u^2 & \Sigma I_u I_v \\ \Sigma I_u I_v & \Sigma I_u^2 \end{pmatrix}, \tag{1}$$

where $I_u$ and $I_v$ are the discrete partial derivatives of the image in the $u$ and $v$ directions accordingly. We observe that the maximal eigenvalue of $S$ corresponds to the strongest directional change around every pixel. Our premise is that features with strong enough directional change, contain valuable information for matching and transformation estimation. Thus our feature selection consists in the following thresholding for each pixel

$$\max(\lambda_1, \lambda_2) > \tau_\lambda, \tag{2}$$

where $\tau_\lambda$ is some fixed threshold and $\{\lambda_1, \lambda_2\}$ are the eigenvalues of $S$. Indeed, one strong directional change doesn't ensure that the feature could be independently well-localized in both dimensions, but only in one. Thus, we call them weakly-localized features. The advantage of this approach over the well-known minimal eigenvalue thresholding [23] is in extracting a larger amount of features for a given threshold. This will allow us to analyze and utilize many more points around $O$, at the expense of having to compensate for the poor localization of some of them. Thus, the candidate locations$\{s^{(k)}\}_{k=1}^{K}$ for matching are those satisfying Equation (2). We note that choosing $\tau_\lambda$ presents a trade-off between potential coverage and lower computational effort. In practice we found that $\tau_\lambda = 0.01$ is low enough to capture the vast majority of matchable features, while keeping the computational demand sufficiently low. In comparison, we found that $\tau_\lambda = 0.1$ eventually produces 50% of the match coverage produced by $\tau_\lambda = 0.01$, while incurring 25% of the computational demand, while $\tau_\lambda = 0.001$ produces 105% of the coverage, but with 200% of the computational demand.

## 2.2 Step 2: Match Prediction

For each location $s^{(k)}$ in the source image, we now wish to locate its corresponding location $t^{(k)}$ in the target image. For this purpose, we can now use our estimation of the affine transform given by $\hat{Q}(x)$, with parameters $\hat{q}$. Following [9], the error co-variance in predicting $t^{(k)}$ using the estimator $\hat{q}$ is given by

$$\Sigma_p^{(k)} = J_Q(x)\Sigma_{\hat{q}} J_Q^\intercal(x)\Big|_{x=s^{(k)}}, \tag{3}$$

where $J_Q\big|_{s^{(k)}} = \frac{\partial Q(x)}{\partial q}\Big|_{s^{(k)}}$ is the Jacobian of the transformation $Q(x)$ with respect to its parameters $q$, evaluated at the point $s^{(k)}$, and $\Sigma_{\hat{q}}$ is the co-variance matrix of the estimator $\hat{q}$. As the expression in Equation (3) is quite general, we wish to apply it specifically for the affine model. For this, we define $[x_{hh}]_{2\times 6} \triangleq (I_{2\times 2} \otimes x_h)^\intercal$, where $x_h = [\begin{array}{cc} x & 1 \end{array}]^T = [u, v, 1]^T$ is the homogenization of a point $[u, v]$, and $\otimes$ is the Kronecker product. For convenience, we vectorize the parameters of the affine model to $[q]_{6\times 1} \triangleq vec([\begin{array}{cc} A & b \end{array}])$. Thus, we have $Q(x) \triangleq Q(x_{hh}) = x_{hh} \cdot q$ as a vectorized representation of the 2D affine model. We now plug-in to Equation (3) and get

$$\Sigma_p^{(k)} = J_Q(x_{hh})\Sigma_{\hat{Q}} J_Q^\intercal(x_{hh})\Big|_{x_{hh}=s_{hh}^{(k)}} \underbrace{=}_{J_Q(x_{hh}) = x_{hh}} x_{hh}\Sigma_{\hat{Q}} x_{hh}^\intercal\Big|_{x_{hh}=s_{hh}^{(k)}} = s_{hh}^{(k)}\Sigma_{\hat{q}}\left(s_{hh}^{(k)}\right)^\intercal. \tag{4}$$

Thus, for any source point $s^{(k)}$, we have a prediction $\hat{t}^{(k)} = \hat{A}s^{(k)} + \hat{b}$ and a corresponding covariance matrix $\Sigma_p^{(k)}$ around $\hat{t}^{(k)}$. To calculate the matrices $\Sigma_p^{(k)}$, we need to have the co-variance of the estimated parameters given by $\Sigma_{\hat{q}}$. For this, we assume that our initial estimation was based on a set of $N$ point matches $\{s0^{(j)}, t0^{(j)}\}_{j=1}^{N}$, where all the target points $\{t0^{(j)}\}$ are estimated to be unbiased, uncorrelated along themselves and along the image axes, and have the same variance $\sigma^2$ in both axes. In formal terms, this assumption translates to having initial correspondence covariance $\Sigma_{\hat{T}} = \sigma^2 \cdot I_{2N\times 2N}$. In [6, 7], the effect of $\sigma$ was analyzed in the context of match prediction and scanning, where it was shown that larger $\sigma$ demand proportionally larger scanning windows, which in turn increases the computational demand. For simplicity, we also assume *zero error of locations in the source image points* $\{s0^{(j)}\}$, which is equivalent to all source points being under our affine model. Thus, we have

$$\Sigma_{\hat{q}} = \sigma^2 \cdot (I_{2\times 2} \otimes S0)^{\intercal} (I_{2\times 2} \otimes S0),$$

where $S0 = \begin{bmatrix} s0_h^{(1)} & s0_h^{(2)} & \dots & s0_h^{(N)} \end{bmatrix}_{3\times N}$ is a concatenation of all homogenized source points used for estimating $\hat{q}$. We note that for classic matching methods such as [15], the variance $\sigma^2$ is generally unknown, but can be estimated for specific data-sets. Plugging back to (4), the $\Sigma_p^{(k)}$'s are now known for each $k$. To conclude this stage, we now have a target point prediction $\hat{t}^{(k)}$ and a corresponding error co-variance $\Sigma_p^{(k)}$ around $\hat{t}^{(k)}$ for each source point $s^{(k)}$ that can serve for the constrained match search phase.

## 2.3 Step 3: Match Scanning

We are now ready to separately refine our estimation for each target location $t^{(k)}$. We define a rectangular search window $W^{(k)}$ around the prediction $\hat{t}^{(k)}$ with dimensions $D_u \times D_v$ that bound the confidence ellipse defined by $\beta \cdot C^{(k)}$, where $\beta$ is a confidence factor. For example, $\beta = 2.45$ yields $\approx 95\%$ confidence in locating $t^{(k)}$ within the ellipse. Thus, in this case, the rectangle corresponds to a 2D confidence interval that includes $t^{(k)}$ with probability at least 0.95. These ellipses and corresponding bounding boxes are illustrated in Figure 4b. Similarly to [6], we use NCC scanning adapted by the estimation $\hat{Q}$ and its inverse $\hat{Q}^{-1}$, in the following manner:

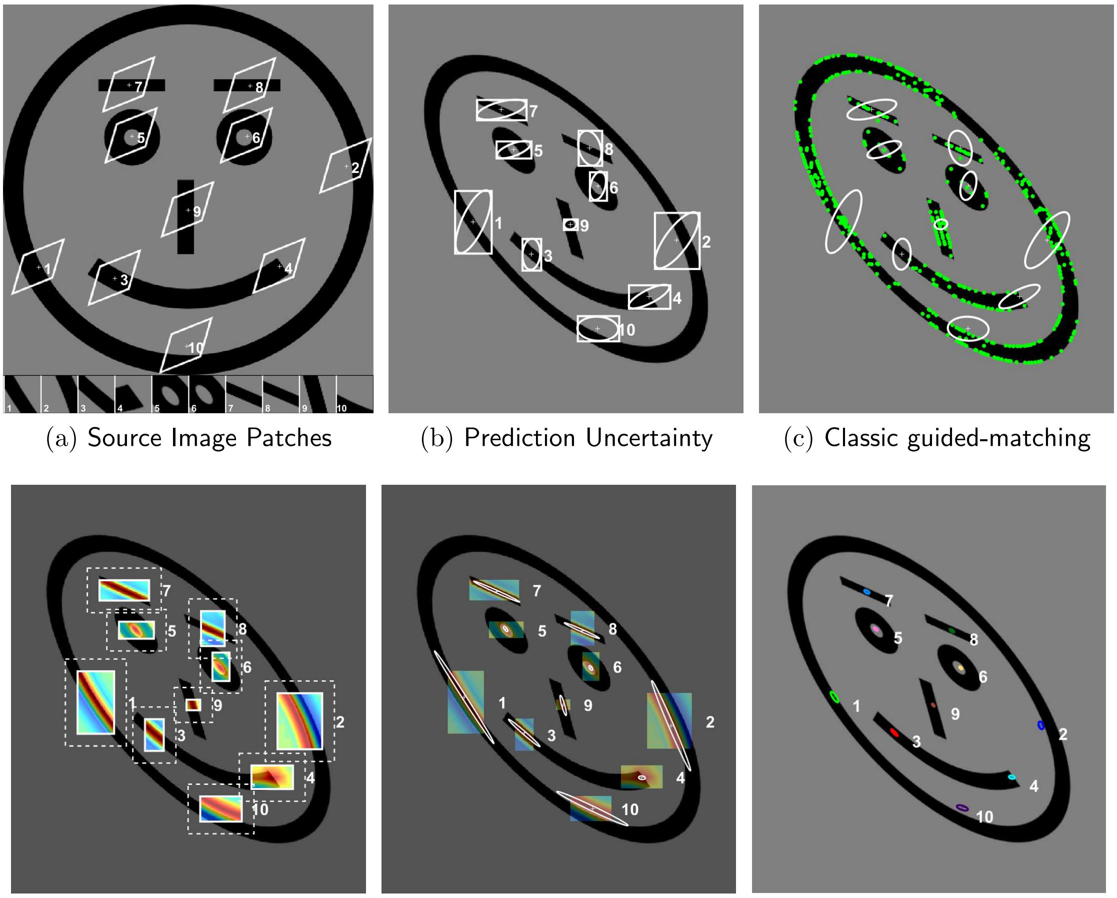
1. Define a $D_u \times D_v$ domain around $\hat{t}^{(k)}$, denoted $w_{tgt}^{(k)}$. (Figure 4d - the dashed rectangles).

2. Render a $D_u \times D_v$ patch using the interpolated image levels (Figure 4a - bottom):

$$\pi^{(k)} = I_{src}(\hat{Q}^{-1}(w_{tgt}^{(k)})). \tag{5}$$

3. Perform $NCC$ on the patch $I_{tgt}(W^{(k)})$ with $\pi^{(k)}$ as a template to get the response $NCC^{(k)}$ (Figure 4d - colored).

We note that the rendered patch given in Equation (5) might not be perfectly produced for optimal template matching, as blur and intense radiometric artifacts are not invertible. Moreover, using a naive interpolation method (we use bilinear) might also contribute to aliasing artifacts. As in [6], we observe that the NCC algorithm is relatively robust to these imperfections, while the consideration of a more accurate patch rendering method is considered out of the scope of this work. The refined estimation of $t^{(k)}$ is then given by the coordinates of the maximal value of $NCC^{(k)}$. As pointed in [6], the refined location can be considered reliable when the correlation response is high (namely, close to 1), and the spatial distribution of the response resembles a 2D Kronecker delta. This corresponds to a well-localized match. As illustrated in Figure 4d, the response is likely to resemble a delta only when the structure tensor of the patch $\pi^{(k)}$ is not degenerate, which corresponds to $s^{(k)}$ being a well-localized key-point or a corner. Indeed, well-localized key-points



(a) Source Image Patches (b) Prediction Uncertainty (c) Classic guided-matching

(d) NCC Results (e) NCC Localization Uncertainty (f) Jointly Reduced Localization Uncertainty

Figure 4: Illustration of the proposed matching and estimation framework (best viewed in color): (4a) - The patches taken around each feature in the source image, and their corresponding adjustments (bottom) to the geometry of target image using the initial transformation estimation; (4b) - The projection uncertainty of the initial transformation estimation represented as confidence ellipses for a given confidence interval (99%), and the corresponding bounding rectified rectangles to be used as scanning windows. (4c)-The classic guided-matching alternative, candidate target matches are located within each corresponding ellipse. (4d) - The result of running NCC with the adjusted templates from (4a) on the target image using the corresponding valid scanning windows (solid rectangles), dashed rectangles account for the entire scanning area. Strong responses represented in dark red; (4e) - The localization uncertainty induced by the corresponding NCC scans, represented as confidence ellipses for a 99% confidence interval; (4f) - The significantly reduced localization uncertainty induced by the refined transformation estimation that jointly takes into account the uncertainties from 4e.

are sparse in the image, thus the set of reliable NCC results is also sparse, leaving us with many weakly-localized matches. Instead of discarding these matches, we wish to take advantage of the information they entail. For this purpose, we observe that although weakly-localized matches might not be reliable as separate matches, combining several such matches under a common model might produce highly reliable information. The most natural common model for all the scanned matches is the affine transformation $Q$. Thus, we wish to formalize the re-estimation of the parameters $q$, taking into account all the NCC scanning results. This should produce two desirable results: 1) the new estimation $\hat{\hat{q}}$ will be more accurate than $\hat{q}$; 2) the new estimation of $t^{(k)}$, given by $\hat{\hat{Q}}(s^{(k)})$, should be more accurate than $\hat{t}^{(k)}$ for every $k$, and be *well-localized* even for locations with weakly-localized scans. In the following sub-section we formalize the estimation technique that can realize these results.

## 2.4 Step 4: Transformation Re-Estimation

First, we separately observe each of the responses $NCC^{(k)}$ (Figure 4d). We wish to treat the response as a spatial distribution of the probability of each location to be the correct $t^{(k)}$. First, we discard all locations where $NCC^{(k)}$ is significantly lower than its maximal value (typically $< 0.75 \cdot \max(NCC^{(k)})$), since these are most probably outliers. Now we use the soft-max operator to get $P^{(k)} = \frac{\exp(NCC^{(k)})}{\Sigma \exp(NCC^{(k)})}$ which is now a probability function above the 2D coordinates in $W^{(k)}$. Generally, the probability functions $P^{(k)}$ can be fully utilized by using Monte-Carlo sampling to produce $L$ possible guesses for each $t^{(k)}$ according to its corresponding $P^{(k)}$, and re-estimating $q$ by a set of $K \times L$ point pairs. For a large enough number of samples $S$, this may lead to a statistically optimal solution for $q$ given the probabilities $P^{(k)}$, but can also be computationally prohibitive, especially if we need to robustly estimate $q$ using iterative random selection methods [8]. To prevent prohibitive computations, we only extract up to second-order statistics from each $P^{(k)}$ to get the estimated expected location $\mu_{\hat{t}^{(k)}}$ and co-variance matrix $\Sigma_{\hat{t}^{(k)}}$ that represents the spatial ambiguity in estimating $t^{(k)}$ (illustrated by the ellipses in Figure 4e). Assuming $q$ is estimated by a linear estimator, these values can now be substituted into a new estimation of $q$ using a generalized linear estimator (GLE), where each $\mu_{s^{(k)}}$ is accordingly plugged as the expected location of $t^{(k)}$, and the co-variance matrix of all the locations is given by the block diagonal

$$\Sigma_{\hat{T}} = \begin{pmatrix} [\Sigma_{\hat{t}^{(1)}}]_{2\times 2} & 0 & \cdots & 0 \\ 0 & [\Sigma_{\hat{t}^{(2)}}]_{2\times 2} & \cdots & 0 \\ \vdots & 0 & \ddots & \vdots \\ 0 & 0 & \cdots & [\Sigma_{\hat{t}^{(k)}}]_{2\times 2} \end{pmatrix}_{2K\times 2K} \tag{6}$$

We note that $\Sigma_{\hat{T}}$ is zero outside the blocks, since we assume that the estimations of $t^{(i)}$ and $t^{(j)}$ are uncorrelated for $i \neq j$. Additionally, the estimation of $\Sigma_{\hat{T}}$ also allows us to estimate the uncertainty of the re-projection of each point in the source image to a point in the target image, using the estimation of $q$ [21, 9]. This enables not only the fixing of weakly-localized matches, but also an accurate derivation of the uncertainty in each of these fixes. This uncertainty is expected to be dramatically smaller for each of the matches, compared to the original uncertainty, as illustrated by the ellipses in Figure 4f relative to those of figures 4b and 4e. To get the GLE for the affine case, we first denote $S_I \triangleq I_{2\times 2} \otimes S$, with $S$ again being the concatenation $\begin{bmatrix} s_h^{(1)} & s_h^{(2)} & \ldots & s_h^{(K)} \end{bmatrix}$. From the Gauss-Markov Theorem, we have that the best linear unbiased estimator for the affine model is then given by

$$\hat{\hat{q}} = [S_I^\intercal \Sigma_{\hat{T}} S_I]^{-1}\, S_I^\intercal \Sigma_{\hat{T}}^{-1} \cdot \mu_{\hat{T}} = \Sigma_{\hat{\hat{q}}} S_I^\intercal \Sigma_{\hat{T}}^{-1} \cdot \mu_{\hat{T}},$$

where $[\mu_{\hat{T}}]_{2K\times 1} = vec\left(\begin{bmatrix} \mu_{\hat{t}}^{(1)} & \mu_{\hat{t}}^{(2)} & \ldots & \mu_{\hat{t}}^{(K)} \end{bmatrix}\right)$ is a concatenation of all the vectorized scanning means of the target points. $\Sigma_{\hat{T}}$ is given by Equation (6), under the assumption of uncorrelated correspondence errors across the different matches. Thus, we obtained a new estimation of $q$ while also utilizing weakly-localized correspondences. We can thus reuse Equation (4), to extract the re-projected confidences, and observe how the weakly-localized matches with covariances $\{\Sigma_{\hat{t}^{(k)}}\}_{k=1}^{K}$turn into strongly localized ones with co-variances $\left\{ J_Q(x)\Sigma_{\hat{\hat{Q}}} J_Q^\intercal(x)\Big|_{x=s^{(k)}} \right\}_{k=1}^{K}$ accordingly. The full formalization of the framework proposed in this section is given in Algorithm 1.

**Algorithm 1:** The local estimation and match detection algorithm for a single affine seed around a point $O$

```
Input:
$I_{src}, I_{tgt}$ - Two gray-scale images
$\hat{q}$- Estimated parameters for the affine mapping between $I_{src}$ and $I_{tgt}$ around the location $O$
$\Sigma_{\hat{q}}$ - Covariance matrix of the estimated parameters $\hat{q}$
Output:
$\{s^{(k)}\}_{k=1}^{K}, \{\hat{t}^{(k)}\}_{k=1}^{K}$ - Point correspondences between $I_{src}$ and $I_{tgt}$

Collect all close points with sufficient texture:
for all $(u, v) with \|(u, v) - O\| < R$ do                    // points in the vicinity of $O$
    $S = structure\_tensor(\Pi_{u,v})$                        // $\Pi_{u,v}$ - a $U \times V$ patch around (u,v)
    $\lambda_{max} \leftarrow \max(eignvalues(S))$
    $K \leftarrow 0, s \leftarrow \varnothing$
    if $\lambda_{max} > \tau_{\lambda}$ then
        $s^{(K)} \leftarrow (u, v)$
        $K \leftarrow K + 1$
for $1 \leq k \leq K$ do
    $[s_{hh}] = (I_{2\times 2} \otimes [\ s \quad 1\ ]^{\intercal})^{\intercal}$
    $\Sigma^{(k)} \leftarrow \left(s_{hh}^{(k)}\right) \Sigma_{\hat{q}} \left(s_{hh}^{(k)}\right)$
    $\hat{t}^{(k)} \leftarrow \hat{Q}(s^{(k)})$
    $w_{tgt}^{(k)} \leftarrow$ a rectangular patch $D_u \times D_v$, bounding the ellipse $\beta \cdot \Sigma^{(k)}$
    $\pi^{(k)} \leftarrow I_{src}(\hat{Q}^{-1}(w_{tgt}^{(k)}))$
    $NCC^{(k)} \leftarrow NCC(I_{tgt}(W^{(k)}), \pi^{(k)})$
    $P_{u,v}^{(k)} \leftarrow \frac{\exp(NCC^{(k)})}{\Sigma \exp(NCC^{(k)})}$
    $\mu_{\hat{t}^{(k)}} \leftarrow \mathbb{E}_{u,v}$
    $\Sigma_{\hat{t}^{(k)}} \leftarrow \Sigma_{u,v}$
$\Sigma_{\hat{T}} \leftarrow blockdiag(\{\Sigma_{\hat{t}^{(k)}}\}_{k=1}^{K})$
$S_I \triangleq I_{2\times 2} \otimes \left[\ s_h^{(1)} \quad s_h^{(2)} \quad \ldots \quad s_h^{(K)}\ \right]$
$\hat{\hat{q}} \leftarrow \Sigma_{\hat{\hat{q}}} S_I^{\intercal} \Sigma_{\hat{T}}^{-1} \cdot \mu_{\hat{T}}$
$\{\hat{\hat{t}}^{(k)}\}_{k=1}^{K} \leftarrow \hat{\hat{Q}}(\{s^{(k)}\}_{k=1}^{K})$
```

# 3 Matching under the Global Perspective Model: Beyond Guided-Matching

Beyond the local application under the affine model, we can derive a similar framework for more global models that can hold for larger portions of $\Omega_{\vec{F}}$ in specific scenarios. Specifically, we focus this section on the 2D perspective homography given by $Q(x) = \frac{H \cdot x_h}{\lambda}$, where $\lambda = H_3 \cdot x_h$ which can serve as global model for scenarios like planar or very distant scenes. Similarly to the affine case, we show how to exploit the applicability of the perspective model for locating point matches which are normally out of reach for standard matching techniques. We assume we are dealing with two images fully or partly related by a single perspective transform ($Q(x)$), and given an initial estimation of it ($\hat{Q}(x)$). Our goal now is to locate more image matches in the entire image. The steps for reaching this goal are detailed in the following sub-sections.

## 3.1 Step 1: Key-Point Extraction and Assignment

Similarly to the affine case, we wish to extract candidate locations (key-points) that can be successfully matched to the target image. In contrast to the local affine case, we now apply this feature extraction on the entire source image. As there is no conceptual difference from the affine model with respect to feature extraction, we can choose the same, *maximal eigenvalue* based feature extractor, as described in Section 2.1.

## 3.2 Step 2: Match Prediction

Similarly to the affine case, for each location $s^{(k)}$ in the source image, we now wish to locate its corresponding location $t^{(k)}$ in the target image. For this purpose, we can now use our estimation of the perspective transform given by $\hat{Q}(x)$, with parameters $\hat{q}$. As in Equation (7), once again we wish to express the covariance of the projection of each source point $s^{(k)}$

$$\Sigma_p^{(k)} = J_Q(x)\Sigma_{\hat{q}}J_Q^\intercal(x)\big|_{x=s^{(k)}} . \tag{7}$$

Only now we wish to derive it specifically for the perspective model. Similarly to the affine case, we assume $Q$ was estimated from $N$ point matches $\{s0^{(j)}, t0^{(j)}\}_{j=1}^N$ with the same error assumptions and known uniform variance $\sigma^2$. Following [14], we define the Jacobian $J_i|_{x=s} \triangleq \frac{\partial Q(x)}{\partial \hat{q}}\Big|_{x=s} = \frac{1}{\lambda_i}\begin{bmatrix} (s_h)^\intercal & 0_{1\times3} & -u(s_h)^\intercal \\ 0_{1\times3} & (s_h)^\intercal & -v(s_h)^\intercal \end{bmatrix}$, where $q$ again denotes the eight parameters of $Q$, and $(u,v)=t$ is the target coordinate corresponding to the source coordinate $s$. Thus, following the assumption that the $N$ point matches are statistically uncorrelated, the co-variance of $\hat{q}$ is approximated by

$$\Sigma_{\hat{q}} \approx (J_Q^\intercal \Sigma_{\tilde{T}}^{-1} J_Q)^\dagger \underbrace{=}_{\text{uniform error}} (\frac{1}{\sigma^2} J_Q^\intercal J_Q)^\dagger \underbrace{=}_{\text{uncorrelated error}} \left(\frac{1}{\sigma^2}\sum_{i=1}^N J_i^\intercal|_{x=s0^{(i)}}\ J_i|_{x=s0^{(i)}}\right)^\dagger , \tag{8}$$

where † denotes the pseudo-inverse of a matrix. We note that $\hat{q}$ is determined up to an arbitrary scale and is thus of size $1\times9$, but with only eight degrees of freedom. Similarly, we have

$$\begin{aligned}\Sigma_p^{(k)} \approx J_Q(x)|_{x=s^{(k)}}\ \Sigma_{\hat{q}}\ J_Q^\intercal(x)\big|_{x=s^{(k)}} &\underbrace{=}_{\text{uncorrelated and uniform error}} \\ &= J_k^T\big|_{x=s^{(k)}} \left(\frac{1}{\sigma^2}\sum_{i=1}^N J_i^\intercal|_{x=s0^{(i)}}\ J_i|_{x=s0^{(i)}}\right)^\dagger J_k|_{x=s^{(k)}},\end{aligned} \tag{9}$$

which gives a first order approximation for the co-variance in predicting each target point $t0^{(k)}$, given the corresponding sampled source point $s^{(k)}$.

## 3.3 Step 3: Match Scanning

As for each predicted target location $t0^{(k)}$ we have its estimated covariance $\Sigma_p^{(k)}$, we can apply the same *NCC* based scanning algorithm as described in Section 2.3 and extract the means $\{\mu_{\hat{t}^{(k)}}\}_{k=1}^K = \{u_{\hat{t}^{(k)}}, v_{\hat{t}^{(k)}}\}_{k=1}^K$ and corresponding co-variance matrices $\{\Sigma_{\hat{t}^{(k)}}\}_{k=1}^K$ of each scanned target point.

### 3.4 Step 4: Transformation Re-Estimation

While the re-estimation of $q$ ($\hat{\hat{q}}$) can be done using different methods with respect to the error minimization metric [14], we opt for a weighted version of the DLT algorithm [14], which minimizes the algebraic estimation error, due to its analytical simplicity in our case. The solution can be estimated from the set of equation pairs

$$\{A_k\hat{\hat{q}}=0\}_{k=1}^{K} \triangleq \{[\Sigma_{\hat{t}^{(k)}}^{-1}]\cdot\begin{bmatrix} (s_h^{(k)})^\intercal & 0_{1\times3} & -\hat{u}^{(k)}(s_h^{(k)})^\intercal \\ 0_{1\times3} & (s_h^{(k)})^\intercal & -\hat{v}^{(k)}(s_h^{(k)})^\intercal \end{bmatrix}\cdot_{2\times9}[\hat{\hat{q}}]_{9\times1}=[0]_{2\times1}\}_{k=1}^{K}. \tag{10}$$

We can now solve for $\hat{\hat{q}}$ by estimating the 1D null space of $A=\begin{bmatrix} A_1^\intercal & \cdots & A_K^\intercal \end{bmatrix}^\intercal$, normally using SVD [14]. Thus, we obtain a new estimation of $q$ while also utilizing weakly-localized correspondences. Similarly to Equation (8), we can also derive the co-variance of the estimation (up to first order approximation) $\hat{\hat{q}}$

$$\Sigma_{\hat{\hat{q}}}\approx(J_Q^\intercal\Sigma_{\hat{T}}^{-1}J_Q)^\dagger \underbrace{=}_{\text{uncorrelated error}} \left(\sum_{k=1}^{K} J_i^\intercal|_{x=S^{(i)}}\,[\Sigma_{\hat{t}^{(k)}}^{-1}]\,J_i|_{x=S^{(i)}}\right)^\dagger. \tag{11}$$

Similarly to the affine case, $\Sigma_{\hat{T}}$ is given by (6), under the assumption of uncorrelated correspondence errors across the different matches. We note the important difference from (8), where we didn't have explicit information about the co-variance of each individual target location. We can thus reuse (9) and have

$$\Sigma_p^{(k)}{=}\sum_{k=1}^{K} J_i^\intercal|_{x=s^{(k)}}\,\Sigma_{\hat{\hat{q}}}\,J_i|_{x=s^{(k)}},$$

which gives us the re-projected confidence of each match. Similarly to the affine case, we should observe how weakly-localized matches with co-variances $\{\Sigma_{\hat{t}^{(k)}}\}_{k=1}^{K}$turn accordingly into strongly localized ones. The full formalization of the framework proposed in this section is given in Algorithm 2.

## 4 A Statistically Sound Outlier Rejection Procedure

The different estimation procedures presented in sub-sections 2.4 to 3.4 assume all the detected matches to be roughly correct and obeying the same geometric model. This assumption can be justified by the fact that all scanned matches were initialized by the same predicted transformation. While this might promise that a high rate of detected matches will still share a joint model, it does not promise that other matches do not exceed this model. This is especially true for the global perspective case, but also for expansion-based match scanning as presented in [7], where newly located matches lie outside of the initial estimation domain. This gives rise to the well-known challenge of robust estimation or outlier rejection. The most common practice for rejecting such outliers is using random selection techniques such as RANSAC [8]. We present a variation of this approach that also takes into account the knowledge we gained about the different error covariances. The re-projection error of an estimated mapping with respect to point match $(s^{(k)},\hat{t}^{(k)})$ is evaluated by

$$e^{(k)}=\hat{\hat{Q}}(s^{(k)})-\hat{t}^{(k)}.$$

Thus, every hypothesized mapping $\hat{\hat{Q}}$ can be evaluated, for instance, by counting the number of 2D errors $e^{(k)}$ that have a Euclidean norm $||e^{(k)}||_2$ below a certain threshold, and all the matches with re-projection error smaller than that threshold will be considered inliers. Despite its usefulness, this expression has two main drawbacks in the context of this work: 1) it is only adequate for

**Algorithm 2:** The global estimation and match detection algorithm under the perspective model

```
Input:
$I_{src}, I_{tgt}$ - Two gray-scale images
$\hat{q}$- Estimated parameters for the perspective mapping between $I_{src}$ and $I_{tgt}$
$\Sigma_{\hat{q}}$ - Covariance matrix of the estimated parameters $\hat{q}$
Output:
$\{s^{(k)}\}_{k=1}^{K}, \{\hat{t}^{(k)}\}_{k=1}^{K}$ - Point correspondences between $I_{src}$ and $I_{tgt}$

Collect all points with sufficient texture
for all $(u, v) \in \Omega_{Isrc}$ do                                          // for all pixels
    $S = structure\_tensor(\Pi_{u,v})$                         // $\Pi_{u,v}$ - a $U \times V$ patch around (u,v)
    $\lambda_{max} \leftarrow \max(eignvalues(S))$
    $K \leftarrow 0, s \leftarrow \varnothing$
    if $\lambda_{max} > \tau_\lambda$ then
        $s^{(K)} \leftarrow (u, v)$
        $K \leftarrow K + 1$
for $1 \leq k \leq K$ do
    $s_h^{(k)} = [\ s^{(k)} \quad 1\ ]^T$
    $J_Q(s^{(k)}) \leftarrow \frac{1}{\lambda_k} \begin{bmatrix} (s_h^{(k)})^\intercal & 0_{1\times3} & -u(s_h^{(k)})^\intercal \\ 0_{1\times3} & (s_h^{(k)})^\intercal & -v(s_h^{(k)}) \end{bmatrix}$
    $\Sigma^{(k)} \leftarrow J_Q(s^{(k)}) \Sigma_{\hat{q}} J_Q^\intercal(s^{(k)})$
    $\hat{t}^{(k)} \leftarrow (s^{(k)})$
    $w_{tgt}^{(k)} \leftarrow$ a rectangular patch $D_u \times D_v$, bounding the ellipse $\beta \cdot \Sigma^{(k)}$
    $\pi^{(k)} \leftarrow I_{src}(\hat{Q}^{-1}(w_{tgt}^{(k)}))$
    $NCC^{(k)} \leftarrow NCC(I_{tgt}(W^{(k)}), \pi^{(k)})$
    $P_{u,v}^{(k)} \leftarrow \frac{\exp(NCC^{(k)})}{\Sigma \exp(NCC^{(k)})}$
    $\mu_{\hat{t}^{(k)}} \leftarrow \mathbb{E}_{u,v}$
    $\Sigma_{\hat{t}^{(k)}} \leftarrow \Sigma_{u,v}$
    $A_k \leftarrow [\Sigma_{\hat{t}^{(k)}}^{-1}] \cdot \begin{bmatrix} (s_h^{(k)})^\intercal & 0_{1\times3} & -\hat{u}^{(k)}(s_h^{(k)})^\intercal \\ 0_{1\times3} & (s_h^{(k)})^\intercal & -\hat{v}^{(k)}(s_h^{(k)})^\intercal \end{bmatrix}$
$A = [\ A_1^\intercal \quad \cdots \quad A_K^\intercal\ ]^\intercal$
$\hat{\hat{q}} \leftarrow nullspace(A)$
$\{\hat{\hat{t}}^{(k)}\}_{k=1}^{K} \leftarrow \hat{\hat{Q}}(\{s^{(k)}\}_{k=1}^{K})$
```

well-localized matches, while weakly-localized matches are expected to have high re-projection error and thus might vote against good hypotheses; 2) it fails to take into account the information we have about the estimation uncertainty of the hypothesis $\hat{\hat{q}}$. In contrast, we define a slightly different re-projection error metric that relies on examination of Mahalanobis alongside Euclidean distances. For this, we define the following Mahalabonis distances

$$|e^{(k)}|_{loc} = \sqrt{(e^{(k)})^\intercal \Sigma_{\hat{t}^{(k)}}^{-1} (e^{(k)})},$$

$$|e^{(k)}|_{proj} = \sqrt{(e^{(k)})^\intercal \Sigma_{\hat{\hat{p}}^{(k)}}^{-1} (e^{(k)})},$$

where $\Sigma_{\hat{p}^{(k)}}$ is the covariance of the projection $\hat{\hat{Q}}(s^{(k)})$. We note how $|e^{(k)}|_{loc}$ measures the deviation of the re-projection error $e^{(k)}$ from the localization covariance $\Sigma^{-1}_{\hat{t}^{(k)}}$, while $|e^{(k)}|_{proj}$ measures the deviation from the projection covariance $\Sigma_{\hat{p}^{(k)}}$. Thus, combining these two measures together should give a measure for the re-projection error that would be sensitive both to the known localization ambiguity of each match and to the known estimation uncertainty of $\hat{\hat{q}}$. Since we work under a positive hypothesis that the match $(s^{(k)}, \hat{t}^{(k)})$ should normally belong to our model, and want to keep both well-localized and weakly-localized matches, we take a lenient approach for all matches and set the error measure to be

$$e^{(k)}_{min} = min(|e^{(k)}|_{loc}, |e^{(k)}|_{proj}).$$

At this stage, we separate between well-localized matches, which have error co-variance matrix $\Sigma_{\hat{t}^{(k)}}$ with both corresponding ellipse axes smaller than some threshold $\tau_{loc}$ (e.g. five pixels), and the remaining weakly-localized matches. Since well-localized matches contain more statistical information and thus are less prone to numerical instability, we accept every well-localized match with $e^{(k)}_{min} < \tau_{m1}$, while for weakly-localized matches we only accept matches with $e^{(k)}_{min} < \tau_{m1} < \tau_{m2}$. Thus, every hypothesized mapping $\hat{\hat{Q}}$ can be evaluated by counting the number of matches obeying these thresholds accordingly, and all the obeying matches can be treated as inliers of $\hat{\hat{Q}}$. Additionally, we also consider matches with Euclidean error $\left\|e^{(k)}\right\| < \tau_{euc}$ as inliers.

In this way, we formalized an outlier rejection procedure that both takes into account all the statistical information we posses, and is stricter for matches that contain ambiguous information. This outlier rejection procedure can be applied identically for the affine, perspective or any other point-to-point model. In Algorithm 3 we formalize this procedure, noting how it can easily combined with Algorithm 2 or Algorithm 1.

# 5 Empirical Evaluation

We evaluate the proposed framework in two aspects. The first is its ability to locate matches after an initial estimation of a perspective homography, where we compare the proposed method to the classic guided-matching scheme [21], and the second is in the contribution of the framework to the affine-expansion mechanism introduced in [6].

## 5.1 Comparison to Classic Guided-Matching under the Perspective Model

We compare the proposed method to classic guided-matching in two main aspects: scene coverage and matching accuracy. The different methods are compared under the accurate coverage criterion, where we follow [22] to define *coverage@T* as the portion of the domain $\Omega_{\overrightarrow{F}}$, covered by "correct" matches. A pixel is defined "correct" when its matching error is below $T$ pixels. Since different applications demand different precision, we present the results while varying $T$. We note that since all the compared methods are not pixel-dense, it is more adequate to consider a pixel *covered* even if a correct match is only present within some distance from it. Following [22], for all referenced methods, we set this distance to be of 10 pixels. That is, a location is considered covered is a match exist within a radius of 10 pixels around it. Since we focus on the perspective homography model, we used the H-Patches dataset (the "Viewpoint" part) [1] - containing images of 59 partly planar scenes imaged from different viewing angles, supplied with the ground truth homographies of a total of 295 image pairs. Image regions outside the main plane of the scene were ignored in all experiments, since they are not annotated with ground-truth mapping.

**Algorithm 3:** The Mahalanobis-based outlier rejection algorithm

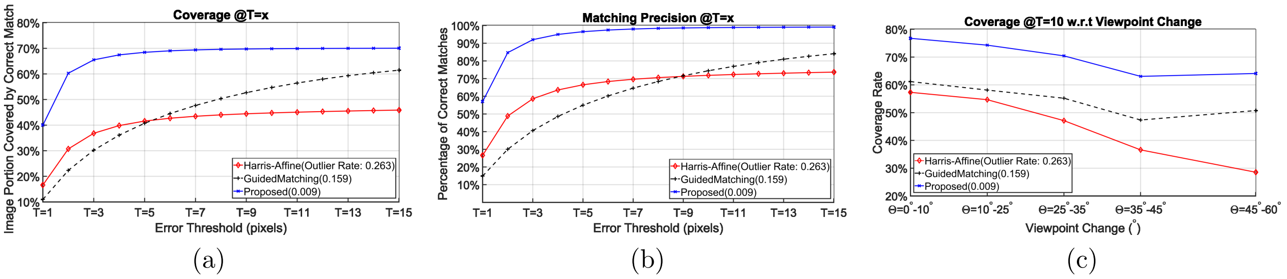
```
Input:
$\{s^{(n)}, \hat{t}^{(n)}\}_{n=1}^{N}$ - Tentative match pairs
$\{\Sigma_{\hat{t}^{(n)}}\}_{n=1}^{N}$ - Corresponding localization covariances
Parameters:
$K$ - Seed size for initial estimation
$NI$ - Minimal amount of required inliers
$\tau_{m1}, \tau_{m2}$ - Inlier Mahalanobis thresholds for well and weakly localized points accordingly
$\tau_e$ - Inlier Euclidean threshold
$GLE$ - The generalized linear estimator according to the geometric model
$PCE$ - The projection error covariance estimator according to the geometric model
Output:
$\{s^{(n)}, \hat{\hat{t}}^{(n)}\}_{n=1}^{N'}$ - Verified match pairs

while $True$ do
    draw K match pairs from $\{s^{(n)}, \hat{t}^{(n)}\}_{n=1}^{N}$
    $\hat{q} \leftarrow GLE(\{s^{(k)}, \hat{t}^{(k)}, \Sigma_{\hat{t}^{(k)}}\}_{k=1}^{K})$
    $\{\Sigma_{\hat{p}^{(n)}}\}_{n=1}^{N} \leftarrow PCE(\{s^{(k)}, \hat{t}^{(k)}\}_{k=1}^{K}, \hat{Q})$
    $\{|e^{(n)}|_{proj} \leftarrow \sqrt{(e^{(n)})^{\intercal} \Sigma_{\hat{p}^{(n)}}^{-1} (e^{(n)})}\}_{n=1}^{N}$
    $\{e_{min}^{(n)} \leftarrow min(|e^{(n)}|_{loc}, |e^{(n)}|_{proj})\}_{n=1}^{N}$
    Inliers_Set $\leftarrow \varnothing$
    for all $n$ with $\|major\_axis(\Sigma_{\hat{t}^{(n)}})\| < \tau_{loc}$ do
        if $e_{min}^{(n)} < \tau_{m1} \vee \|e^{(k)}\| < \tau_{euc}$ then
            Inliers_Set $\leftarrow$ Inliers_Set $\cup \{s^{(n)}, \hat{t}^{(n)}, \Sigma_{\hat{t}^{(n)}}\}$
    for all $n$ with $\|major\_axis(\Sigma_{\hat{t}^{(n)}})\| \geq \tau_{loc}$ do
        if $e_{min}^{(n)} < \tau_{m1} \vee \|e^{(k)}\| < \tau_{euc}$ then
            Inliers_Set $\leftarrow$ Inliers_Set $\cup \{s^{(n)}, \hat{t}^{(n)}, \Sigma_{\hat{t}^{(n)}}\}$
    if $—Inliers\_Set— \geq NI$ then
        break
$\hat{\hat{q}} \leftarrow$GLE(Inliers_Set)                    // re-estimate on inliers
$N' = |Inliers\_Set|$
$\{s^{(n)}, \hat{t}^{(n)}, \Sigma_{\hat{t}^{(n)}}\}_{n=1}^{N'}$ = Inliers_Set
$\hat{\hat{q}} \leftarrow$GLE(Inliers_Set)
$\{\hat{\hat{t}}^{(n)}\}_{n=1}^{N'} \leftarrow \hat{\hat{Q}}(\{s^{(n)}\}_{n=1}^{N'})$      // re-project the estimation to get more reliable matches
```

### Guided-Matching

We examine the matching performance of the proposed method relative to the classic guided-matching approach given in [21]. For both compared methods, we start off by matching each pair of images with the Harris-affine method [18], as implemented in [24]. We use these initial matches to robustly estimate the perspective homography between the images, using RANSAC [8]. To perform guided-matching, we use this initial estimation to predict the locations of all the detected Harris-affine features in the target image. We then use Equation (9), to estimate the error co-variance of each predicted location. For this, we assume that the localization error of the Harris-affine matches is unbiased, and need to have knowledge about its standard deviation, $\sigma$. Following the statistics from [6], we assume $\sigma = 5$, corresponding to a localization error with standard deviation of five pixels on both axes, for each Harris-affine match chosen as an inlier for the robust estimation. Thus, for every predicted match, we can define the search ellipse for guided-matching as a confidence interval that captures 95% of correctly predicted target image locations. This corresponds to an ellipse with roughly *six* times the area of the ellipse representing each prediction covariance $\Sigma^{(k)}$, with the same orientation. Thus, for every source feature, we have a bounded search region, where we can look for its best match in the target image (see Figure 4c for a simple illustration of the search ellipse). As observed in Figure 5, this procedure significantly increases the coverage of correct matches relative to the initial Harris-affine matching, while also increasing the matching precision. This result is well expected, since the guided-matching mechanism uses model information that is not available for the initial Harris-affine matching. This information allows to reduce the rate of inliers and exploit even less repeatable features as appropriate match candidates.

### The Proposed Matching Mechanism

Following the procedure described in Section 3, we extract candidate features in the source image (with corner detection window $U \times V = 17 \times 17$, and max eigenvalue threshold $\tau_\lambda = 0.01$) and predict their target location using the initially estimated perspective homography. We then utilize Equation (9) to define a scanning window for each predicted match. Each window is defined as the bounding rectangle of the 95% ($\beta = 2.45$) confidence ellipse (see Figure 4b for illustration). As described in 3, we then apply the NCC algorithm to refine these predictions and extract their new localization uncertainties in the form of $2 \times 2$ error co-variance matrices. Matches with NCC score lower than 0.5 are discarded. Using Equation (10) and Equation (11) respectively, we use these refined locations and uncertainties to re-estimate the perspective transformation and its error co-variance in a non-robust fashion. We then use the new estimation to re-project the source points back to the target image, and observe their *re-projection error*. Following sub-section 4, we separate between well-localized matches that have error co-variance matrix with both corresponding ellipse axes smaller than five pixels ($\tau_{loc} < 5$), and the remaining weakly-localized matches. We set the threshold for well-localized matches to a Mahalanobis distance of $\tau_{m1} = 2.45$ (for $\approx 95\%$ acceptance confidence interval), while setting the threshold for weakly-localized matches to a more strict $\tau_{m2} = 1.18$ (for $\approx 50\%$ acceptance confidence interval). Target locations that do fall within these Mahalanobis thresholds, or within the Euclidean threshold $\tau_e = 2.5$, are considered as inliers, and their location is corrected to be the re-projected target point. Thus, we now have a set of matches that is a mixture of independently well-localized matches, and matches that are only well-localized given the joint perspective estimation. Following Algorithm 3, this procedure is carried out in a robust fashion with a seed size of $K = max(8, N)$, and requiring at least 80% of the matches to be inliers ($NI \geq 0.8 * N$). If Algorithm 3 doesn't terminate after some iteration limit, we apply the estimation procedure in a non-robust fashion with all the correspondences. In Figure 5, we observe how the proposed model dramatically increases the coverage and precision of matches compared to guided-matching under all benchmark thresholds, due its ability to include weakly-localized features in a controlled manner. This is nicely illustrated in examples given in Figure 6 where the proposed method locates many successful matches in weakly textured image regions. We observe how these matches are beyond the reach of the standard guided-matching approach which relies only on well-localized features. For reference, in Figure 5c, we observe how both the guided-matching and the proposed approach suffer less coverage degradation from increasing the viewpoint change between the image pairs, as they directly model the perspective mapping and are thus less prone to model mismatch compared to the initial affine matching (i.e. Harris-affine).



(a) (b) (c)

Figure 5: Matching Performance: (5a) – Successful coverage of the source image, achieved by the compared methods while varying the allowed error on the H-Patches data-set. (5b) – Rate of correct matches with different error standards on the H-patches dataset. Errors $>$ 15 pixels considered as outliers. (5c) – Successful coverage of the source image as a function of different viewpoint variation scenarios on the H-patches dataset.

## 5.2 Contribution to Affine-Expansion

As described in Section 2, the proposed matching mechanism under the affine model highly resembles the affine-expansion mechanism proposed in [6]. In this context, the main contribution of this work is in replacing the "Delta-Criterion" (DC) approach, focused on extracting only well-localized matches, with a more systematic approach for extracting more information from NCC scans. For this purpose, we follow the procedure and the corresponding equations laid down in Section 2, for handling the affine model. We start by extracting tentative local matches along with their corresponding estimated affine transformations using the Harris-affine method [18]. We separate our analysis across the different matched features and follow Algorithm 1, for each match. Similarly to [6], we set the expansion radius ($R$) to be proportional to the initial extent of the matched feature with an expansion factor of $\alpha = 1.5$. Around the feature in the source image we extract candidate points in the source image (with corner detection window $U \times V = 17 \times 17$, and max eigenvalue threshold $\tau_\lambda = 0.01$). For efficiency, we avoid assigning a candidate point to more than one tentative feature. We then define a scanning window for each predicted match as the bounding rectangle of the 95% ($\beta = 2.45$) confidence ellipse. We then apply the NCC algorithm to refine these predictions and extract their new localization uncertainties. As in Section 5.1, matches with NCC score lower than 0.5 are discarded. We then estimate the affine transform using the new uncertainties. For every scanned match, we observe the corresponding re-projection error of the estimated transform, and its co-variance matrix to discard matches whose re-projection error exceeds the 95% confidence ellipse. Similarly to [6], if less than four matches remain, the entire region match is considered false. Otherwise, the region match is considered verified and we re-estimate the affine transform for it, using the remaining point matches. Following Algorithm 1, we set the target points as the re-projection of the corresponding source points using the estimated affine transform. Similarly to [6], this procedure can be repeated several times to further expand the affine region and locate more point matches around it. For the experiments listed below, we iterate this procedure five times in total for each tentative feature, with expansion factors of $[\alpha = 1.5, 2, 2, 2, 2]$ accordingly.

## Point Matching Quality

In Figure 7, we compare the proposed approach to the DC approach on the H-Patches dataset (the "Viewpoint" part). For further reference, we also include the results for the initial matching method (in this case, Harris-affine [18]). In Figure 7a, we show how the DC approach from [6] indeed dramatically increases the precision of the initial Harris-affine matching. We observe how the precision of the proposed method falls only slightly behind DC, despite dropping the restrictive criterion for well-localized matches. In Figure 7b, we observe how the restrictive DC approach

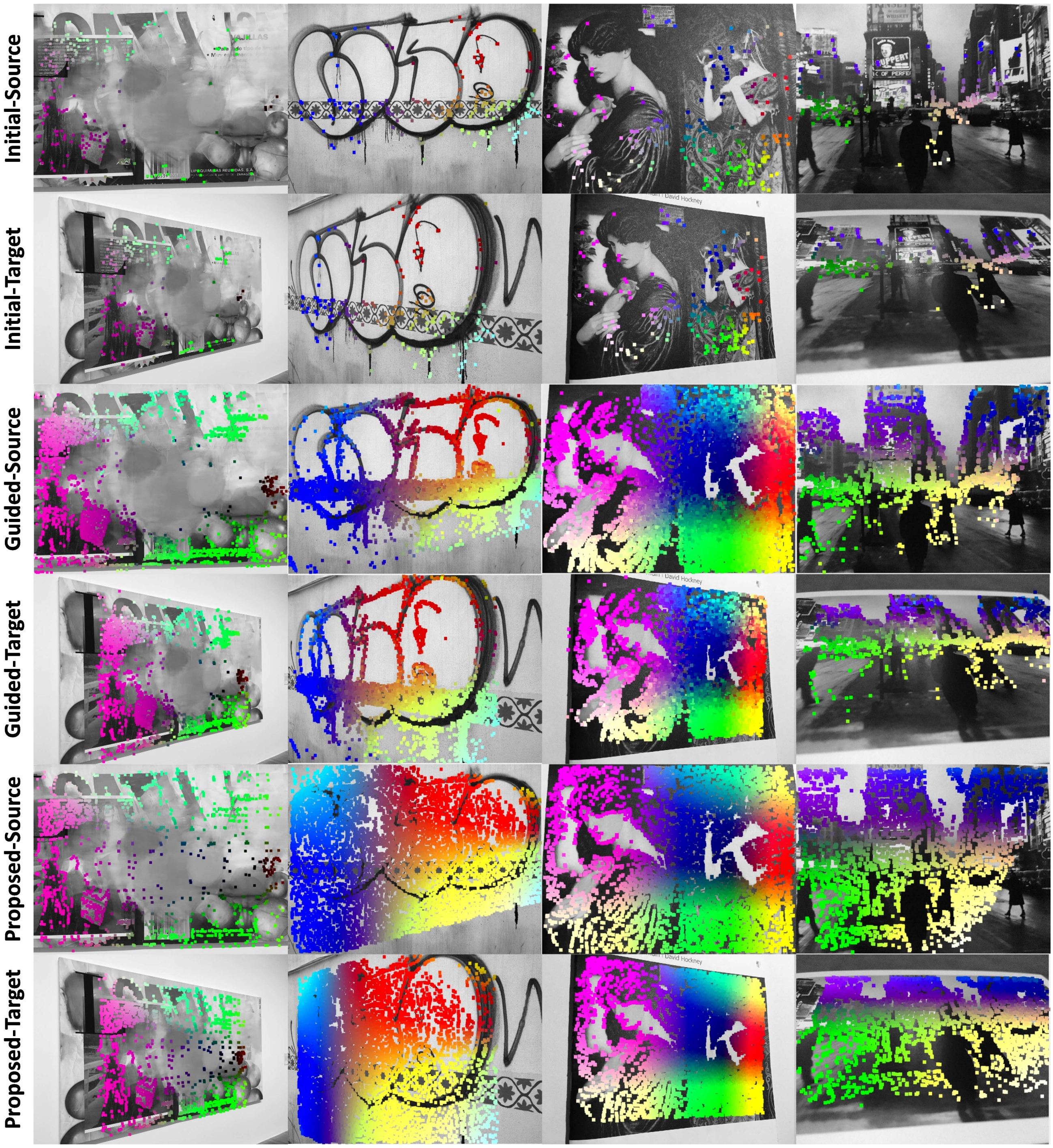
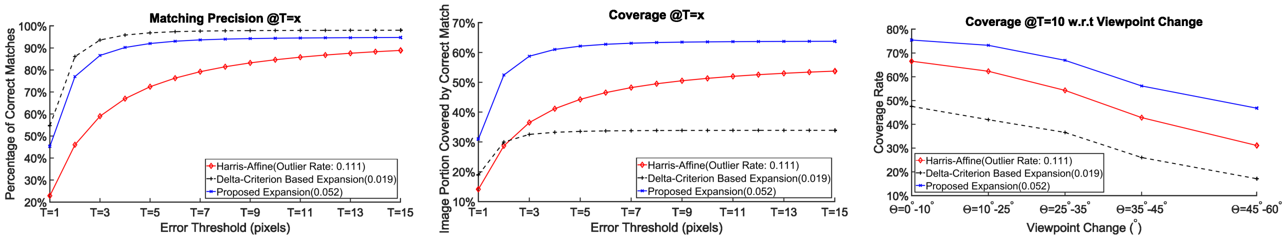


Figure 6: Example results on the H-Patches data-set under the Perspective model assumption: guided-matching expectedly finds matches in highly textured regions, while the proposed method increases match density in low-textured regions.

practically reduces the coverage with respect to the initial Harris-affine matching, as the allowed error threshold increases. This reflects the all-or-nothing nature of well-localized matching restrictions. This behavior is well illustrated in Figure 8 where the limited coverage of the DC based coverage is observed. The proposed method does not suffer from these restrictions, and thus takes better advantage of the expansion mechanism to cover significantly larger portions of $\Omega_{\vec{F}}$ with highly accurate matches. For reference, in Figure 7c, we show how the coverage of all compared methods tends to decrease as the viewpoint variations between the image pairs increase. Indeed, we observe a direct proportion between the coverage of the initial matching method and that of the expansion methods, while the increasing model mismatch between the increasing perspective effects and the affine model limits the effect of local affine expansion.



(a) (b) (c)

Figure 7: Matching Performance: (7b) – Successful coverage of the source image, achieved by the compared methods while varying the allowed error on the H-Patches data-set. (7a) – Rate of correct matches with different error standards on the H-patches dataset. Errors > 15 pixels considered as outliers

# 6 Conclusions

In this work, we have shown how to increase the magnitude and coverage of matched image regions beyond highly-textured features, into ill-textured and weakly-localized domains. We have described a framework for systematically incorporating the uncertainties inherent in weakly-localized matches into a statistically attractive transformation estimation procedure. The practical attractiveness of the proposed framework is exhibited under both the affine and perspective homography models, dramatically increasing match coverage beyond comparable methods while maintaining very high precision.

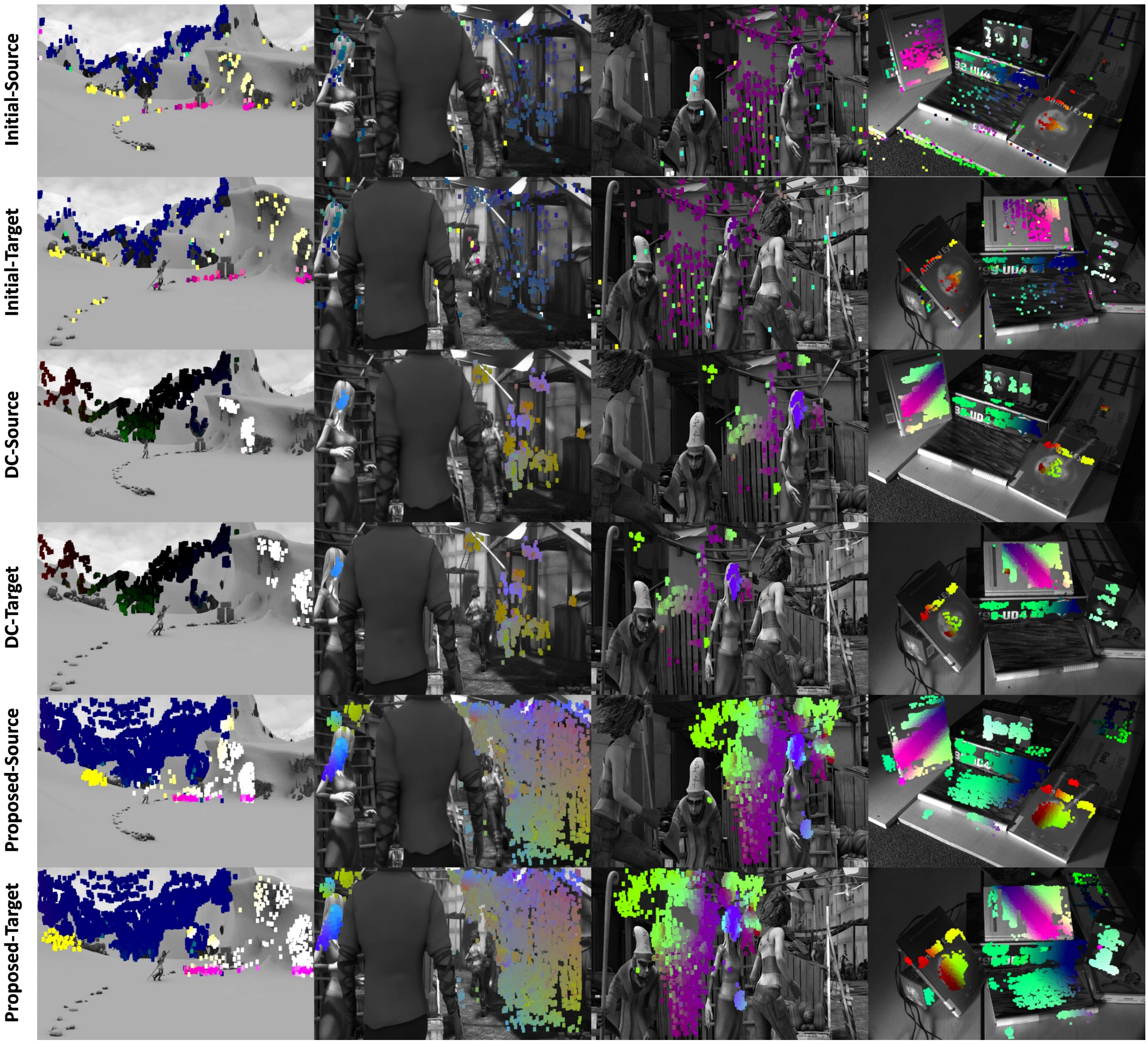


Figure 8: Example results achieved by compared affine-expansion methods: The Delta-Criterion (DC) based expansion successfully rejects many false matches produced by initial matching, while locally densifying around most correct matches. The proposed method maintains the accuracy of the Delta-Criterion, while increasing the density and coverage in less textured areas.

# Image Credits

All images in this manuscript were taken either from the "Affine Covariant Regions" dataset[2], or produced by the authors (license CC-BY-SA).

# References

[1] V. Balntas, K. Lenc, A. Vedaldi, and K. Mikolajczyk, *HPatches: A benchmark and evaluation of handcrafted and learned local descriptors*, in IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2017. https://doi.org/10.1109/CVPR.2017.410.

[2] H. Bay, T. Tuytelaars, and L. Van Gool, *SURF: Speeded up robust features*, in Computer Vision–ECCV 2006, Springer, 2006, pp. 404–417. https://doi.org/10.1007/11744023_32.

[3] J. Bentolila and J.M. Francos, *Homography and fundamental matrix estimation from region matches using an affine error metric*, Journal of Mathematical Imaging and Vision, 49 (2014), pp. 481–491. https://doi.org/10.1007/s10851-013-0481-0.

[4] M. Calonder, V. Lepetit, C. Strecha, and P. Fua, *BRIEF: Binary robust independent elementary features*, in Computer Vision–ECCV 2010, Springer, 2010, pp. 778–792. https://doi.org/10.1007/978-3-642-15561-1_56.

[5] E. Farhan and R. Hagege, *Affine estimation via region expansion*, in IEEE Workshop on Statistical Signal Processing (SSP), IEEE, 2014, pp. 81–84. https://doi.org/10.1109/SSP.2014.6884580.

[6] ——, *Geometric expansion for local feature analysis and matching*, SIAM Journal on Imaging Sciences, 8 (2015), pp. 2771–2813. https://doi.org/10.1137/140997671.

[7] E. Farhan, E. Meir, and R. Hagege, *Local region expansion: a method for analyzing and refining image matches*, Image Processing On Line, 7 (2017), pp. 386–398. https://doi.org/10.5201/ipol.2017.154.

[8] M.A. Fischler and R.C. Bolles, *Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography*, Communications of the ACM, 24 (1981), pp. 381–395. https://doi.org/10.1145/358669.358692.

[9] W. Förstner, *Uncertainty and projective geometry*, in Handbook of Geometric Computing, Springer, 2005, pp. 493–534. https://doi.org/10.1007/3-540-28247-5_15.

[10] ——, *Minimal representations for uncertainty and estimation in projective spaces*, in Asian Conference on Computer Vision, Springer, 2010, pp. 619–632. https://doi.org/10.1007/978-3-642-19309-5_48.

[11] D. Fortun, P. Bouthemy, and C. Kervrann, *Optical flow modeling and computation: a survey*, Computer Vision and Image Understanding, 134 (2015), pp. 1–21. https://doi.org/10.1016/j.cviu.2015.02.008.

[12] ——, *Aggregation of local parametric candidates with exemplar-based occlusion handling for optical flow*, Computer Vision and Image Understanding, 145 (2016), pp. 81–94. https://doi.org/10.1016/j.cviu.2015.11.020.

[2] https://www.robots.ox.ac.uk/~vgg/data/affine/

[13] C. Harris and M. Stephens, *A combined corner and edge detector*, in Alvey Vision Conference, vol. 15, Manchester, UK, 1988, p. 50.

[14] R. Hartley and A. Zisserman, *Multiple view geometry in computer vision*, Cambridge university press, 2003.

[15] D.G. Lowe, *Distinctive image features from scale-invariant keypoints*, International Journal of Computer Vision, 60 (2004), pp. 91–110. https://doi.org/10.1023/B:VISI.0000029664.99615.94.

[16] J. Matas, O. Chum, M. Urban, and T. Pajdla, *Robust wide-baseline stereo from maximally stable extremal regions*, Image and Vision Computing, 22 (2004), pp. 761–767. https://doi.org/10.1016/j.imavis.2004.02.006.

[17] J. Meidow, C. Beder, and W. Förstner, *Reasoning with uncertain points, straight lines, and straight line segments in 2D*, ISPRS Journal of Photogrammetry and Remote Sensing, 64 (2009), pp. 125–139. https://doi.org/10.1016/j.isprsjprs.2008.09.013.

[18] K. Mikolajczyk and C. Schmid, *Scale & affine invariant interest point detectors*, International Journal of Computer Vision, 60 (2004), pp. 63–86. https://doi.org/10.1023/B:VISI.0000027790.02288.f2.

[19] J-M. Morel and G. Yu, *ASIFT: A new framework for fully affine invariant image comparison*, SIAM Journal on Imaging Sciences, 2 (2009), pp. 438–469. https://doi.org/10.1137/080732730.

[20] R. Mur-Artal, J.M.M. Montiel, and J.D. Tardós, *ORB-SLAM: a versatile and accurate monocular SLAM system*, IEEE Transactions on Robotics, 31 (2015), pp. 1147–1163. https://doi.org/10.1109/TRO.2015.2463671.

[21] B. Ochoa and S. Belongie, *Covariance propagation for guided matching*, in Workshop on Statistical Methods in Multi-Image and Video Processing (SMVP), vol. 83, 2006.

[22] J. Revaud, P. Weinzaepfel, Z. Harchaoui, and C. Schmid, *Deepmatching: Hierarchical deformable dense matching*, International Journal of Computer Vision, 120 (2016), pp. 300–323. https://doi.org/10.1007/s11263-016-0908-3.

[23] C. Tomasi and J. Shi, *Good features to track*, in IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 1994, pp. 593–600. https://doi.org/10.1109/CVPR.1994.323794.

[24] A. Vedaldi and B. Fulkerson, *VLFeat: An open and portable library of computer vision algorithms*, in 18th ACM international conference on Multimedia, MM '10, New York, NY, USA, 2010, ACM, ACM, pp. 1469–1472.

[25] T. Whelan, S. Leutenegger, R. Salas-Moreno, B. Glocker, and A. Davison, *ElasticFusion: Dense SLAM without a pose graph*, Robotics: Science and Systems, 2015. https://doi.org/10.15607/RSS.2015.XI.001.

[26] J. Yang and H. Li, *Dense, accurate optical flow estimation with piecewise parametric model*, in IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2015, pp. 1019–1027. https://doi.org/10.1109/CVPR.2015.7298704.

[27] J. YANG, J. WRIGHT, T.S. HUANG, AND Y. MA, *Image super-resolution via sparse representation*, IEEE Transactions on Image Processing, 19 (2010), pp. 2861–2873. https://doi.org/10.1109/TIP.2010.2050625.